



Profils d'unité archivistique

Date	Version
15/06/2018	2.0 (Release 7)

État du document

En projet Vérifié Validé

Maîtrise du document

Responsabilité	Nom	Entité	Date
Rédaction	MVI	Équipe Vitam	04/04/2018
Vérification	Équipe	Équipe Vitam	
Validation		Équipe Vitam	15/06/18

Suivi des modifications

Version	Date	Auteur	Modifications
0.1	04/04/2018	MVI	Initialisation
1.0	05/06/2018	EVA	Relecture
1.1	06/06/18	MRE	Relecture
2.0	15/06/2018	MRE	Finalisation du document pour publication de la Release 7

Documents de référence

Document	Date de la version	Remarques
NF Z44022 – MEDONA - Modélisation des données pour l'archivage	18/01/2014	
Standard d'échange de données pour l'archivage – SEDA – v. 2.1	06/2018	
Vitam - Structuration des <i>Submission Information Package (SIP)</i> – v. 4.0.	20/03/2018	

Licence

La solution logicielle VITAM est publiée sous la licence CeCILL 2.1 ; la documentation associée (comprenant le présent document) est publiée sous Licence Ouverte V2.0.

Table des matières

1. Résumé.....	4
1.1 Présentation du programme Vitam.....	4
1.2 Présentation du document.....	5
2. Présentation de la notion de profil d’unité archivistique.....	6
2.1. Qu’est-ce qu’un profil d’unité archivistique ?.....	6
2.2. Pourquoi élaborer un profil d’unité archivistique ?.....	6
3. Formalisation des profils d’unité archivistique.....	9
3.1. Dans un fichier propre.....	9
3.2. Dans le Standard d’échange de données pour l’archivage (SEDA).....	9
4. Mécanismes mis en œuvre dans la solution logicielle Vitam.....	10
4.1. Administration d’un référentiel des profils d’unité archivistique.....	10
4.1.1. Import de la notice descriptive du profil d’unité archivistique.....	10
4.1.2. Ajout du profil d’unité archivistique lui-même.....	11
4.1.3. Modification de la notice descriptive d’un profil d’unité archivistique.....	12
4.1.4. Activation / Désactivation d’un profil d’unité archivistique.....	12
4.2. Processus d’entrée.....	13
4.3. Accès.....	14
4.4. Mise à jour des unités archivistiques.....	15
4.4.1. Déclaration <i>a posteriori</i> d’un profil d’unité archivistique.....	15
4.4.2. Modification des métadonnées d’une unité archivistique déclarant un profil d’unité archivistique.....	15
5. Étapes d’élaboration d’un profil d’unité archivistique et de son schéma de contrôle.....	16
5.1. Analyse des données.....	16
5.2. Schématisation des données à contrôler.....	17
5.2.1. Rédaction d’un schéma de contrôle.....	18
5.2.2. Comment exprimer les besoins métier dans un schéma de contrôle.....	31
5.2.3. Validation et documentation d’un schéma de contrôle.....	32
5.2.4. Corrections d’un profil d’unité archivistique.....	32
5.2.5. Documentation d’un profil d’unité archivistique.....	32
5.2.6. Réalisation d’un bordereau de transfert conforme au profil d’unité archivistique.....	32
6. Conseils de mise en œuvre.....	33
Annexe 1 : Schéma de contrôle par défaut dans la solution logicielle Vitam (état au 16/05/2018).....	34
Annexe 2 : Exemple de notice détaillée d’un profil d’unité archivistique.....	75
Annexe 3 : Messages d’erreur.....	76

1. Résumé

Jusqu'à présent, pour la gestion, la conservation, la préservation et la consultation des archives numériques, les acteurs du secteur public étatique ont utilisé des techniques d'archivage classiques, adaptées aux volumes limités dont la prise en charge leur était proposée. Cette situation évolue désormais rapidement et les acteurs du secteur public étatique doivent se mettre en capacité de traiter les volumes croissants d'archives numériques qui doivent être archivés, grâce à un saut technologique.

1.1 Présentation du programme Vitam

Les trois ministères (Armées, Culture et Europe et Affaires étrangères), combinant légalement mission d'archivage définitif et expertise archivistique associée, ont décidé d'unir leurs efforts, sous le pilotage de la Direction interministérielle du numérique et du système d'information et de communication de l'État (DINSIC), pour faire face à ces enjeux. Ils ont décidé de lancer un programme nommé Vitam (Valeurs Immatérielles Transmises aux Archives Pour Mémoire) qui couvre plus précisément les opérations suivantes :

- la conception, la réalisation et la maintenance mutualisées d'une solution logicielle d'archivage électronique de type back-office, permettant la prise en charge, le traitement, la conservation et l'accès aux volumes croissants d'archives (projet de solution logicielle Vitam) ;
- l'intégration par chacun des trois ministères porteurs du Programme de la solution logicielle dans sa plate-forme d'archivage. Ceci implique l'adaptation ou le remplacement des applications métiers existantes des services d'archives pour unifier la gestion et l'accès aux archives, la reprise des données archivées depuis le début des années 1980, la réalisation d'interfaces entre les applications productrices d'archives et la plate-forme d'archivage (projets SAPHIR au MEAE, ADAMANT au MC et ArchiPél au MinArm) ;
- le développement, par un maximum d'acteurs de la sphère publique, de politiques et de plates-formes d'archivage utilisant la solution logicielle (projet Ad-Essor).

La solution logicielle Vitam est développée en logiciel libre et recourt aux technologies innovantes du Big Data, seules à même de relever le défi de l'archivage du nombre d'objets numériques qui seront produits ces prochaines années par les administrations de l'État. Afin de s'assurer de la qualité du logiciel livré et de limiter les décalages calendaires de réalisation, le projet est mené selon une conduite de projet Agile. Cette méthode dite « itérative », « incrémentale » et « adaptative » opère par successions de cycles réguliers et fréquents de développements-tests-corrections-intégration. Elle associe les utilisateurs tout au long des développements en leur faisant tester les éléments logiciels produits et surtout en leur demandant un avis sur la qualité des résultats obtenus. Ces contrôles réguliers permettent d'éviter de mauvaises surprises lors de la livraison finale de la solution logicielle en

corrigeant au fur et à mesure d'éventuels dysfonctionnements.

Le programme Vitam bénéficie du soutien du Commissariat général à l'investissement dans le cadre de l'action : « Transition numérique de l'État et modernisation de l'action publique » du Programme d'investissement d'avenir. Il a été lancé officiellement le 9 mars 2015, suite à la signature de deux conventions, la première entre les ministères porteurs et les services du Premier ministre, pilote du programme au travers de la DINSIC, et la seconde entre les services du Premier ministre et la Caisse des dépôts et consignations, relative à la gestion des crédits attribués au titre du Programme d'investissements d'avenir.

1.2 Présentation du document

Le document présente les fonctionnalités associées à la prise en compte de la notion de profil d'unité archivistique dans la solution logicielle Vitam.

Il s'articule autour des axes suivants :

- une présentation de la notion de profil d'unité archivistique ;
- une présentation de la manière dont le Standard d'échanges de données pour l'archivage (SEDA) la formalise ;
- une présentation des mécanismes mis en œuvre dans la solution logicielle Vitam pour prendre en compte cette notion, en application du SEDA ;
- une présentation des modalités de création d'un profil d'unité archivistique et de son schéma de contrôle ;
- des recommandations aux utilisateurs de la solution logicielle Vitam sur l'élaboration d'un profil d'unité archivistique (*partie à rédiger*) ;
- quelques conseils complémentaires de mise en œuvre (*partie à rédiger*).

Le présent document décrit les fonctionnalités qui sont offertes par la solution logicielle Vitam au terme de la *release 7* (juin 2018). Il a vocation à être amendé, complété et enrichi au fur et à mesure de la réalisation de la solution logicielle Vitam et des retours et commentaires formulés par les ministères porteurs et les partenaires du programme.

2. Présentation de la notion de profil d’unité archivistique

2.1. Qu’est-ce qu’un profil d’unité archivistique ?

Le profil d’unité archivistique permet de définir la syntaxe, la sémantique et les règles de gestion applicables à une **unité archivistique**. Il constitue une déclinaison du modèle de description propre au Standard d’échanges de données pour l’archivage (SEDA) à l’échelle d’une unité archivistique.

En d’autres termes, pour une unité archivistique, il précise :

- les éléments de description (ou vocabulaires) attendus, ainsi que leur cardinalité,
- le cas échéant, les valeurs requises,
- les règles de gestion qui lui sont associées.

À titre d’exemple, un bordereau de transfert conforme au SEDA et non associé à un profil d’archivage ou à des profils d’unité archivistique n’aura ses métadonnées contrôlées que par rapport au schéma par défaut associé au SEDA et pourra présenter une grande hétérogénéité de description, même pour une même catégorie d’archives : photographies numériques, dossiers de subventions, comptes-rendus de réunion, etc. En revanche, un bordereau de transfert pour lequel des profils d’unité archivistique auront été définis disposera d’unités archivistiques dont la description sera uniforme et propre à un type d’archives ou à une typologie documentaire en particulier, par exemple : des photographies numériques.

Le profil d’unité archivistique est spécifique à une unité archivistique. Mais, contrairement au profil d’archivage qui impose une structuration au bordereau de transfert, un profil d’unité archivistique peut être utilisé dans différents contextes de transfert, issus de services producteurs différents.

Il peut également être utilisé dans un profil d’archivage pour spécifier des types d’unité archivistique, en plus du contrôle effectué par le profil d’archivage.

Il accompagne l’unité archivistique tout au long de sa prise en charge par le système d’archivage électronique en permettant de contrôler ses métadonnées aussi bien à l’entrée que lors de toute modification.

2.2. Pourquoi élaborer un profil d’unité archivistique ?

Comme pour un profil d’archivage, définir un profil d’unité archivistique permet de :

- faciliter, voire automatiser, les contrôles sur les transferts d’archives (à l’échelle des unités archivistiques) d’un service producteur à un service d’archives, depuis un service externe vers la solution logicielle Vitam ;
- effectuer un contrôle supplémentaire sur les transferts reçus et leur contenu, s’ajoutant à l’ensemble des tâches visant à s’assurer de la conformité des données reçues par rapport à celles qui ont été transférées. Toutes ces actions sont gages de qualité des données, au moment de leur réception dans un système d’archivage électronique ;

- définir la sémantique et la syntaxe attendue pour des unités archivistiques à transférer, permettant ainsi une meilleure connaissance de celles-ci et la rédaction de spécifications sur les lots à exporter à fournir à un prestataire externe (tiers-archiviste, éditeur de solutions logicielles, etc.) ;
- imposer un modèle de description identique sur une typologie particulière d’archives, produites dans différents contextes.

Contrairement au profil d’archivage, les profils d’unité archivistique accompagnent les unités archivistiques tout au long de leur prise en charge dans la solution logicielle Vitam et permettent de contrôler les données, leur sémantique et leur syntaxe, lors d’une éventuelle mise à jour des unités archivistiques ou de leurs règles de gestion associées.

2.3. Quand et comment élaborer un profil d’unité archivistique ?

L’élaboration d’un profil d’unité archivistique précède l’ouverture d’un transfert à destination d’un système d’archivage électronique.

Elle peut être envisagée dans les cas suivants :

- transferts de flux applicatifs, afin de les automatiser ;
- transferts de dossiers ou de documents dits « sériels », obéissant strictement à des règles de classement, de nommage et de description uniforme (par exemple, des images numérisées par un service d’archives) ;
- transferts réguliers et récurrents d’un même type d’archives et donc volonté de disposer d’une description homogène, y compris pour faciliter les recherches sur celles-ci.

Dans ces cas-là, son utilisation peut entrer en concurrence avec celle d’un profil d’archivage.

En revanche, contrairement au profil d’archivage, qui requiert de connaître la hiérarchie des données et leur structuration, un profil d’unité archivistique peut être utilisé pour des types ou catégories d’archives dont la structuration précise ne peut être définie par défaut (absence de connaissance a priori du nombre de niveaux dans l’arborescence) et qui sont :

- des documents bureautiques, pris isolément,
- structurées et décrites de manière non uniforme (par exemple, des boîtes mail).

Un profil d’unité archivistique peut s’utiliser indifféremment dans différents contextes. Par exemple, un profil détaillant un modèle d’« acte de naissance » peut être utilisé pour un transfert d’archives d’un SIRH, mais également d’une application de demande de retraite. Il peut alors répondre à un besoin plus transverse de contrôle et de gestion des données dans la solution logicielle Vitam.

La rédaction d’un profil d’unité archivistique, en fonction de l’usage que l’on souhaite en faire, suppose un travail préparatoire, adapté à chaque lot ou catégorie d’archives concerné

(identification des documents à transférer en tenant compte des calendriers de conservation ou des circulaires de tri existants, détermination des métadonnées utiles à la recherche, structuration de l’information dans chaque transfert, etc.).

Elle résulte d’un travail entre les différents acteurs définis dans le Standard d’échange de données pour l’archivage (SEDA), à savoir :

- le service d’archives (administrateurs fonctionnels et techniques du système d’archivage électronique, archivistes en charge de la collecte),
- le service producteur (responsables de l’administration fonctionnelle et technique de l’application concernée quand il s’agit de l’archivage d’une application),
- le service de contrôle quand il existe.

3. Formalisation des profils d’unité archivistique

3.1. Dans un fichier propre

Les profils d’unité archivistique prennent la forme de fichiers au format JSON. En termes de syntaxe et de grammaire, ils correspondent plus exactement à des **schémas de contrôle**, permettant de valider des contenus exprimés au format JSON, qui est le format de structuration des données utilisé dans la solution logicielle Vitam.

Ces profils d’unité archivistique doivent se conformer :

- aux règles émises par le **Standard d’échange de données pour l’archivage (SEDA)**. Ainsi, ils peuvent :
 - émettre des règles plus strictes que celles du SEDA (ex. : rendre unique un titre, alors que le standard rend possible sa répétabilité), mais pas des règles moins strictes (ex : rendre répétable le niveau de description n’est pas conforme au SEDA) ;
 - rendre obligatoire l’utilisation de certains éléments et/ou valeurs ;
- aux vocabulaires définis dans l’**ontologie**. Ils doivent en effet définir éléments et types compatibles avec ceux référencés dans l’ontologie.

3.2. Dans le Standard d’échange de données pour l’archivage (SEDA)

La manière de déclarer un profil d’unité archivistique est définie dans la norme NF Z44-022 et dans sa déclinaison pour les acteurs du service public, le Standard d’échanges de données pour l’archivage (SEDA).

La norme NF Z44-022 offre la possibilité de déclarer un profil d’unité archivistique dans un bordereau de transfert, au moment de la demande de transfert à un service d’archives (message ArchiveTransfer, bloc DataObjectPackage) dans les blocs **ArchiveUnit** pour chaque unité archivistique qui le nécessite.

Dans le sous-bloc **ArchivalUnitProfile**, est indiqué l’identifiant du profil d’unité archivistique correspondant à l’unité archivistique que l’on souhaite contrôler, tel qu’il figure dans le référentiel des profils d’unité archivage (dit des documents type) de la solution logicielle Vitam. Cet identifiant est celui de la notice détaillée d’un profil d’unité archivistique.

4. Mécanismes mis en œuvre dans la solution logicielle Vitam

La solution logicielle Vitam offre à un service d’archives ou à un service externe plusieurs fonctionnalités lui permettant de mettre en œuvre des profils d’unité archivistique :

- l’**administration d’un référentiel** des profils d’unité archivistique ;
- en entrée du système, une **vérification que les unités archivistiques** présentes dans le SIP et déclarant un profil d’unité archivistique **sont conformes à celui-ci** ;
- lors de la mise à jour des unités archivistiques, la déclaration a posteriori d’un profil d’unité archivistique et la **vérification que les unités archivistiques modifiées** et déclarant un profil d’unité archivistique **restent conformes** à celui-ci

4.1. Administration d’un référentiel des profils d’unité archivistique

La solution logicielle Vitam intègre un référentiel des profils d’unité archivistique, administrable par un utilisateur doté des droits adéquats (**administrateur fonctionnel**) et géré dans une collection particulière¹.

Ce référentiel est propre à chaque tenant de la solution logicielle Vitam.

Il se compose de notices descriptives au format JSON, qui incluent nativement un schéma de contrôle au format JSON ou auxquelles il est possible d’associer *a posteriori* un schéma.

Il est possible de réaliser les opérations présentées ci-dessous.

4.1.1. Import de la notice descriptive du profil d’unité archivistique

Il est possible d’importer 1 à n notice(s) descriptive(s) de profil(s) d’unité archivage sous la forme d’un fichier JSON.

Il s’agit d’une opération d’administration.

Cette notice descriptive comprend les informations suivantes pour un profil d’unité archivistique donné :

- identifiant système unique de la notice correspondant à un GUID (`_id`) attribué automatiquement par la solution logicielle Vitam ;
- identifiant (Identifiant) de la notice qui peut être donné au profil d’unité archivistique, généré automatiquement par la solution logicielle Vitam ou par l’application à l’origine de sa création. S’il est généré par la solution logicielle Vitam, cet identifiant se compose du préfixe AUP, suivi d’un tiret et d’une suite de 6 chiffres incrémentés automatiquement ;
- nom du profil d’unité archivistique, qui doit être obligatoirement renseigné et unique dans la solution logicielle Vitam (Name) ;

¹ Pour plus d’informations sur la modélisation de cette collection, consulter le document *Modèle de données*, chapitre « Collection ArchiveUnitProfile ».

- description du profil d’unité archivistique, obligatoire (Description) ;
- statut « Actif » ou « Inactif » (Status) ;
- schéma de contrôle, obligatoire (ControlSchema).
 - Il est destiné à contenir l’ensemble des éléments qui feront l’objet d’un contrôle en entrée ou en mise à jour ;
 - Les éléments qu’il contient doivent se conformer aux vocabulaires définis dans l’ontologie, que ce soit en termes de nommage ou de typage ;
- date de création de la notice descriptive du profil d’unité archivistique (CreationDate) ;
- dernière date de modification de la notice descriptive du profil d’unité archivistique (LastUpdate) ;
- si la notice descriptive du profil d’unité archivistique est actif, date d’activation du profil (ActivationDate) ;
- si la notice descriptive du profil d’unité archivistique est inactif, date de désactivation du profil (DeactivationDate) ;
- tenant dans lequel le profil d’unité archivistique s’applique (_tenant) ;
- version du profil d’archivage (_v).

Points d’attention :

- Il est possible d’importer, en une seule fois, un référentiel complet, comprenant plusieurs notices décrivant chacune un profil d’unité archivistique. La solution logicielle Vitam ne comptabilisera qu’une seule opération, et ne prendra pas en compte dans le journal des opérations la création unitaire des différents items compris dans le référentiel importé ;
Afin d’optimiser la traçabilité de la création des différentes notices, il est recommandé de créer ces dernières une par une.
- La solution logicielle Vitam permet d’importer une notice détaillée avec un schéma de contrôle déjà détaillé ou vide de tout contenu. Dans ce dernier cas, l’import d’une notice ne suffit pas pour utiliser un profil d’unité archivistique dans la solution logicielle Vitam. Il est nécessaire, après cette opération, d’ajouter le schéma de contrôle correspondant au profil d’unité archivistique lui-même et de l’associer à la notice qui le référence, afin qu’il soit pris en compte dans la solution logicielle Vitam (chapitre 4.1.2.).

4.1.2. Ajout du profil d’unité archivistique lui-même

À chaque notice descriptive, il est possible d’associer un seul profil d’unité archivistique ou schéma de contrôle.

Dès son import, la notice détaillée peut inclure un schéma de contrôle détaillé, dans la mesure où tous deux sont au format JSON.

Néanmoins, elle peut être importée sans schéma détaillé. Il est alors possible, *a posteriori*, de lui adjoindre ce schéma de contrôle et de vérifier sa validité par rapport au format JSON.

L’ajout d’un schéma de contrôle dans la collection est un des prérequis indispensable pour pouvoir réaliser des contrôles entre un profil d’unité archivistique et des unités archivistiques présentes dans des bordereaux de transfert au moment de leur transfert dans la solution logicielle Vitam ou déjà prises en charge dans la solution logicielle Vitam et en cours de modification.

Il s’agit alors d’une opération de mise à jour.

4.1.3. Modification de la notice descriptive d’un profil d’unité archivistique

La modification des champs des notices décrivant les profils d’unité archivistique est possible au moyen des API et de l’IHM standard fournie avec la solution logicielle Vitam.

Cette action provoque la création d’une nouvelle version de la notice modifiée. Elle fait l’objet d’une journalisation dans le journal des opérations.

Les champs modifiables sont :

- le nom du profil d’unité archivistique, qui doit rester unique dans la solution logicielle Vitam (Name) ;
- la description du profil d’archivage (Description) ;
- le statut « Actif » ou « Inactif » (Status) ;
- le schéma de contrôle (ControlSchema).

Points d’attention :

- le statut de la notice descriptive doit être « Actif » pour pouvoir procéder à des transferts de SIP comprenant des unités archivistiques devant se conformer à un profil d’unité archivistique (service non implémenté en *release 7*) ;
- la solution logicielle Vitam permet de modifier le profil d’unité archivistique ou son schéma de contrôle, tant que ce dernier n’est pas utilisé par une unité archivistique en entrée ou déjà transférée dans la solution logicielle Vitam. S’il l’est, il n’est plus possible de le modifier.

4.1.4. Activation / Désactivation d’un profil d’unité archivistique

La solution logicielle Vitam permet de rendre active ou inactive une notice détaillant un profil d’unité archivistique.

En fonction du statut, seront autorisés ou non :

- le transfert de SIP dont au moins une unité archivistique déclare un profil d’unité archivistique ;
- la modification d’une unité archivistique déclarant un profil d’unité archivistique.

	Catégorie	Profil d’unité	Schéma de	Résultat
--	-----------	----------------	-----------	----------

	d’opération	archivistique	contrôle	
CAS 1	Transfert	ACTIF	vide	Transfert de SIP dans le système non autorisé.
CAS 2		ACTIF	renseigné	Transfert de SIP dans le système autorisé.
CAS 3		INACTIF	vide	Transfert de SIP dans le système non autorisé.
CAS 4		INACTIF	renseigné	Transfert de SIP dans le système non autorisé.
CAS 5	Modification	ACTIF	vide	Modification d’unité archivistique non autorisé.
CAS 6		ACTIF	renseigné	Modification d’unité archivistique autorisé.
CAS 7		INACTIF	vide	Modification d’unité archivistique non autorisé.
CAS 8		INACTIF	renseigné	Modification d’unité archivistique non autorisé.

La modification du statut engendre la mise à jour des champs :

- Date de mise à jour ;
- Date d'activation OU date de désactivation.

La date d'activation correspond à la date à laquelle la notice et, par conséquent, le profil d’unité archivistique sont rendus actifs. Il peut s'agir de :

- sa date d'import, si elle a un statut « Actif » au moment de son import ;
- la date correspondant à l'action d'activation, si celle-ci est postérieure à l’import de la notice dans la solution logicielle Vitam.

La date de désactivation correspond à la date à laquelle la notice et par conséquent le profil d’unité archivistique sont rendus inactifs. Il peut s'agir de :

- sa date d'import, si elle a un statut « Inactif » ou non renseigné au moment de son import ;
- la date correspondant à l'action de désactivation, si celle-ci est postérieure à l’import de la notice dans la solution logicielle Vitam.

4.2. Processus d’entrée

Dans le cadre du processus d’entrée d’un ensemble d’archives, suite à la réception d’un message ArchiveTransfer du SEDA, parmi les tâches et traitements internes qu’elle effectue, la solution logicielle Vitam **vérifie que les unités archivistiques déclarant un profil d’unité archivistique sont conformes à ce dernier.**

Ce contrôle s’effectue lors du processus de contrôle et traitements des unités archivistiques,

avant les vérifications portant sur les règles de classification et de gestion.

Lors de cette étape de vérification de la conformité entre le profil d’unité archivistique déclaré dans au moins une unité archivistique présente dans le bordereau de transfert :

- Si l’unité archivistique est conforme à son profil d’unité archivistique, si elle correspond au modèle de données défini dans le profil d’unité archivistique, la tâche de vérification aura un statut « OK » et la solution logicielle Vitam passera à la tâche suivante, de vérification du niveau de classification ;
- Si au moins une unité archivistique n’est pas conforme à son profil d’unité archivistique pour les raisons suivantes :
 - elle ne correspond pas au profil d’unité archivistique qu’elle déclare,
 - le profil d’unité archivistique déclaré est inconnu,
 - le profil d’unité archivistique déclaré est inactif,
 - la notice descriptive du profil d’unité archivistique est active, mais son schéma de contrôle est vide, (à vérifier)

Alors le transfert du SIP échouera à la tâche de vérification de conformité aux profils d’unités archivistiques, passera à la tâche suivante, mais n’aboutira pas. Un message de réponse (ArchiveTransferReply ou ATR) indiquera que le transfert a échoué à la tâche de vérification de la conformité au profil d’unité archivistique et précise la première erreur rencontrée dans le détail de l’événement ([EventDetailData](#)).

Exemple : message d’erreur sur la tâche de contrôle de conformité au document type.

```
<ArchiveUnit id="ID3">
  <Management>
    <LogBook>
      <Event>
        <EventTypeCode>LFC.CHECK_ARCHIVE_UNIT_PROFILE</EventTypeCode>
        <EventType>Vérification de la conformité aux documents type</EventType>
        <EventDateTime>2018-04-27T10:37:30.404</EventDateTime>
        <Outcome>KO</Outcome>
        <OutcomeDetail>LFC.CHECK_ARCHIVE_UNIT_PROFILE.KO</OutcomeDetail>
        <OutcomeDetailMessage>Échec de la vérification de la conformité aux
          documents type</OutcomeDetailMessage>
        <EventDetailData>{"evDetTechData":"Archive Unit Profile not
          found"}</EventDetailData>
      </Event>
    </LogBook>
  </Management>
  <Content>
    <SystemId>aeaqaabaabehc7dlhabspualda2w5tlyaaaba</SystemId>
  </Content>
</ArchiveUnit>
```

4.3. Accès

Par défaut, lorsque l'utilisateur demande à accéder au détail d'une unité archivistique, la solution logicielle Vitam retourne l'identifiant du profil d'unité archivistique associé à l'unité archivistique.

4.4. Mise à jour des unités archivistiques

4.4.1. Déclaration *a posteriori* d'un profil d'unité archivistique

En mise à jour, la solution logicielle Vitam permet :

- d'associer un profil d'unité archivistique à une unité archivistique préalablement transférée dans le système ;
- de modifier le profil d'unité archivistique associé à une unité archivistique.

Cette opération de mise à jour n'est possible qu'aux conditions suivantes :

- l'unité archivistique doit être conforme au profil d'unité archivistique à laquelle on souhaite l'associer ;
- le profil d'unité archivistique doit être actif et disposer d'un schéma de contrôle détaillé.

Si l'une de ces conditions n'est pas présente, la modification n'aura pas lieu.

4.4.2. Modification des métadonnées d'une unité archivistique déclarant un profil d'unité archivistique

La solution logicielle Vitam permet de modifier les métadonnées d'une unité archivistique déclarant un profil d'unité archivistique, à la condition que l'unité archivistique reste conforme au profil d'unité archivistique à laquelle elle on souhaite l'associer.

5. Étapes d’élaboration d’un profil d’unité archivistique et de son schéma de contrôle

5.1. Analyse des données

Un profil d’unité archivistique nécessite une analyse préalable des données à archiver, type de document par type de document.

Il s’agit de connaître :

- les types de documents rencontrés (ex : documents bureautiques, factures, mails, etc.) ;
- les métadonnées associées à ces données, permettant de les identifier, de les gérer et de les retrouver après transfert dans le système d’archivage électronique (métadonnées de pérennisation, de représentation et de description au sens de la norme OAIS) ;
- le contenu de données, à savoir la nature des données à archiver (catégorie ou typologie d’archives).

En prenant l’exemple d’un projet de dématérialisation d’archives, il est recommandé, lors de ce travail préparatoire, de se poser les questions suivantes² :

Identifier le contexte	Quels sont les documents à archiver ? Pourquoi ? Existe-il un besoin légal ? La reprise de l’existant a-t-elle été prévue ?
Identifier les documents dématérialisés	Quelles sont les typologies concernées ? Quels sont les usages ? Les documents et données sont-ils nativement numériques ou ont-ils été numérisés ? Une impression papier est-elle prévue en fin de procédure ? De quelle(s) base(s) de données les données et documents sont-ils issus ? Quels sont les formats des documents ? Quelles sont les possibilités d’exports et le modèle des données ? Les documents ont-ils une valeur probante (ex. signature électronique) ?
Identifier les outils techniques	Dans le cadre d’une numérisation, s’agit-il d’une opération interne ? Le recours à un prestataire externe est-il prévu ? Une reprise du stock a-t-elle été envisagée ? L’application a-t-elle été développée en interne ? L’acquisition d’un logiciel est-elle prévue ? Quelles métadonnées ont été intégrées ? Quelles métadonnées peut-on extraire de l’application ? L’organisation a-t-elle recours à un tiers de télétransmission ? Un parapheur électronique est-il en place ? Les agents disposent-ils de la signature électronique ? Est-elle systématiquement utilisée ?
Stockage, archivage et sort final	Quelles sont les règles de gestion et plus particulièrement les règles de conservation et le sort final des documents ? Comment sont conservées les pièces dématérialisées ? Où ? Par qui (organisation, tiers archiveur) ? Existe-t-il un système d’archivage électronique ? Un coffre-fort ? Des serveurs répliqués ? Un système de gestion électronique de documents ?

² Le tableau suivant est une libre adaptation du tableau présenté dans « Fiche pratique n°13 : positionnement de l’archiviste face à un projet de dématérialisation », Association des Archivistes de France.

	Quelles sont les modalités d’accès ? La destruction est-elle possible ? Quelles sont les modalités ?
--	---

A l’issue de cette première analyse, l’archiviste doit être en mesure d’avoir à sa disposition :

- ,
- une liste des métadonnées liées à chaque type de document ;
- les règles de gestion associées.

Le résultat de ce travail peut prendre la forme du tableau suivant :

Type de document ou d’unité archivistique	Métadonnées		Contenu de données
	Informations de description (métadonnées)	Règles de gestion	Documents versés en pièces jointes (oui/non)

La liste des métadonnées peut prendre la forme du tableau suivant :

Nom de la métadonnée dans le système d’origine	type	cardinalité	Obligatoire ?	Valeurs imposées	Commentaires	À récupérer ?
	Texte, Date, Chiffres, etc.	1-1, 0-1, 0-n	OUI / NON	Ex : liste, valeur héritée.		OUI / NON
Age du capitaine	Numérique	0-1	OUI	Entier entre 0 et 150		OUI

Il est nécessaire de bien évaluer les informations à retenir dans le profil d’unité archivistique, leur degré de précision (formes, valeurs, cardinalités), afin de faciliter les recherches et la gestion futures des données dans la solution logicielle Vitam.

5.2. Schématisation des données à contrôler

Suite à ce travail préliminaire, il faut ensuite réaliser un schéma de contrôle associé à un profil d’unité archivistique, puis préparer un bordereau de transfert contenant une unité archivistique conforme à ce profil d’unité archivistique afin de pouvoir réaliser des tests.

Il est conseillé de procéder par étape :

- création du schéma de contrôle depuis un éditeur JSON,
- le cas échéant, en parallèle, documentation du schéma de contrôle sous forme de tableur ou de document,
- génération d’un bordereau conforme au SEDA, contenant une unité archivistique conforme au profil d’unité archivistique et à son schéma préalablement réalisés.

5.2.1. Rédaction d’un schéma de contrôle

Actuellement, il n’existe pas d’outil pour générer un schéma de contrôle associé à un profil d’unité archivistique au format JSON.

Pour en rédiger un, il faut créer un fichier dans un éditeur de texte (ex : Notepad ++, Oxygen) et l’enregistrer en tant que fichier JSON.

Un schéma de contrôle associé à un profil d’unité archivistique comprend deux parties :

- une partie déclarative ou **en-tête** (obligatoire),
- une partie listant les **propriétés** du schéma de contrôle (obligatoire), c’est-à-dire énumérant les éléments du SEDA ou de l’ontologie qui feront l’objet d’un contrôle. Le contenu de la liste est laissé à l’appréciation de chacun.

Cette partie peut elle-même être décomposée en deux ou trois sous-parties :

- contexte d’utilisation de l’unité archivistique,
- règles de gestion à contrôler au moyen du profil d’unité archivistique (équivalent au contenu du bloc Management du SEDA),
- énumération des métadonnées descriptives à contrôler (équivalent au contenu du bloc Content du SEDA).

5.2.1.1. En-tête d’un schéma de contrôle

Comme dans un fichier au format XML, le schéma de contrôle JSON comporte des éléments d’information propres au fichier, permettant :

- d’identifier son format :
 - **\$schema** : référence au schéma JSON,
 - **id** : URI du schéma JSON, permettant de contrôler la validité du schéma de contrôle associé à un profil d’unité archivistique par rapport au format JSON,
 - **type** : toujours égal à « objet » dans l’en-tête. Le type « objet » est assimilable à un dictionnaire, possédant des propriétés. En langage XML, il correspondrait à un élément englobant des sous-éléments et ne pouvant contenir de valeurs.
 - **properties** : dictionnaire listant les propriétés associées au schéma de contrôle, et, de fait, au profil d’unité archivistique. On y énumère les éléments à contrôler.

```
{
"$schema": "http://vitam-json-schema.org/draft-04/schema#",
"id": "http://example.com/root.json",
"type": "object",
```

```
"properties": {  
  .....  
}
```

- de déterminer des règles propres aux objets (notion de "type": "object" JSON) à appliquer aux propriétés qui vont être définies dans le schéma :

- **minProperties** : entier indiquant le nombre minimum d’éléments attendus dans une unité archivistique.

S’il n’est pas renseigné, le nombre minimum d’éléments attendus est égal à 0.

- **maxProperties** : entier indiquant le nombre maximum d’éléments attendus dans une unité archivistique.

S’il n’est pas renseigné, le nombre maximum d’éléments attendus est égal à 0.

Pour se conformer au profil, l’unité archivistique devra comprendre entre 2 et 3 éléments. Si elle n’en contient qu’un, par exemple Title, elle ne sera pas conforme au profil.

```
{  
  "$schema": "http://vitam-json-schema.org/draft-04/schema#",  
  "id": "http://example.com/root.json",  
  "type": "object",  
  "minProperties": 2,  
  "maxProperties": 3,  
  "properties": {  
    "_id": {"type": "string"},  
    "_og": {"type": "string"},  
    "Title": {"type": "string"}  
  }  
}
```

- **additionalProperties** :

- booléen :

- Si sa valeur est égale à « true », une unité archivistique pourra contenir des éléments (métadonnées) supplémentaires, non définis dans le schéma de contrôle,
- Si sa valeur est égale à « false », une unité archivistique contenant des éléments supplémentaires, non définis dans le schéma de contrôle, ne sera pas conforme à son profil et ne pourra pas être versée dans la solution logicielle Vitam ;

Pour se conformer au profil, l’unité archivistique devra contenir les éléments SystemId, DataObjectReference et Title. En revanche, il ne pourra pas contenir d’autres éléments tels que Description ou Keyword, car additionalProperties est égal à false.

```
{  
  "$schema": "http://vitam-json-schema.org/draft-04/schema#",  
  "id": "http://example.com/root.json",  
  "type": "object",  
  "additionalProperties": false,  
  "properties": {  
    "_id": {"type": "string"},  
  }  
}
```

```
"_og": {"type": "string"},
"Title": {"type": "string"}
}
}
```

- liste permettant de préciser et contrôler les éléments qui ne sont pas définis dans la liste des propriétés du schéma de contrôle.

Le profil d'unité archivistique permet à l'unité archivistique le déclarant de contenir des éléments qu'il n'a pas défini. Néanmoins, ces éléments additionnels doivent obligatoirement être de type texte (= string), sans quoi l'unité archivistique est rejetée.

```
{
"$schema": "http://vitam-json-schema.org/draft-04/schema#",
"id": "http://example.com/root.json",
"type": "object",
"additionalProperties": {"type": "string"},
"properties": {
  "_id": {"type": "string"},
  "_og": {"type": "string"},
  "Title": {"type": "string"}
}
}
```

- **dependencies** : liste, permettant de définir des relations de dépendance entre éléments.
 - Si l'élément défini en premier est présent, alors l'élément qui le suit doit l'être également.
 - La dépendance n'est pas réciproque. Pour rendre obligatoire le premier élément si l'élément suivant est présent, il faudra préciser que, si ce dernier est présent, le premier élément doit l'être.

Dans l'exemple suivant, pour se conformer au profil, l'unité archivistique devra contenir les éléments SystemId et Title. Si l'un des deux est présent, l'autre doit l'être obligatoirement.

```
{
"$schema": "http://vitam-json-schema.org/draft-04/schema#",
"id": "http://example.com/root.json",
"type": "object",
"dependencies": {
  "_id": ["Title"],
  "Title": ["_id"]
}
"properties": {
  "_id": {"type": "string"},
  "Title": {"type": "string"}
}
}
```

- de déterminer des règles génériques à appliquer aux propriétés qui vont être définies dans le schéma :
 - **required** : liste d'éléments de type texte (= string) obligatoires. Ces éléments doivent ensuite être définis en tant que propriétés du schéma de contrôle.

S’il s’agit de sous-propriétés (ex : KeywordContent, sous-élément et sous-propriété du bloc Keyword), la règle ne s’appliquera pas.

- **anyOf** : liste imposant qu’au moins l’une des configurations déclarées soit vérifiée, quelle qu’elle soit.

Dans l’exemple suivant, pour se conformer au profil, l’unité archivistique devra contenir soit les éléments SystemId et Title, soit le seul DataObjectReference, soit les trois éléments ensemble.

```
{
  "$schema": "http://vitam-json-schema.org/draft-04/schema#",
  "id": "http://example.com/root.json",
  "type": "object",
  "additionalProperties": false,
  "anyOf": [
    {
      "required": ["_og"],
      "required": ["_id", "Title"]
    }
  ],
  "properties": {
    "_id": {"type": "string"},
    "_og": {"type": "string"},
    "Title": {"type": "string"}
  }
}
```

- **allOf** : liste d’éléments devant être vérifiés. Tous les éléments doivent faire l’objet d’un contrôle.

L’unité archivistique devra contenir obligatoirement les éléments SystemId, Title et DataObjectReference. Si l’un d’eux manque, il ne sera pas conforme au profil d’unité archivistique.

```
{
  "$schema": "http://vitam-json-schema.org/draft-04/schema#",
  "id": "http://example.com/root.json",
  "type": "object",
  "additionalProperties": false,
  "allOf": [
    {
      "required": ["_og"],
      "required": ["_id", "Title"]
    }
  ],
  "properties": {
    "_id": {"type": "string"},
    "_og": {"type": "string"},
    "Title": {"type": "string"}
  }
}
```

- **oneOf** : liste imposant que les données soient exactement conformes à l’une des configurations proposées.

Le schéma de contrôle impose que l’unité archivistique contienne soit un SystemId et un Title, soit un DataObjectReference. En aucun cas, elle ne pourra contenir les trois éléments ensemble.

```
{
  "$schema": "http://vitam-json-schema.org/draft-04/schema#",
  "id": "http://example.com/root.json",
  "type": "object",
  "additionalProperties": false,
  "oneOf": [
    {
      "required": ["_og"],
      "required": ["_id", "Title"]
    }
  ],
  "properties": {
    "_id": {"type": "string"},
    "_og": {"type": "string"},
    "Title": {"type": "string"}
  }
}
```

- **not** : liste déclarant des éléments ne devant pas être présents dans l’unité archivistique déclarant le profil d’unité archivistique ou excluant un paramétrage en particulier.

Le schéma de contrôle interdit que l’unité archivistique contienne un DataObjectReference.

```
{
  "$schema": "http://vitam-json-schema.org/draft-04/schema#",
  "id": "http://example.com/root.json",
  "type": "object",
  "additionalProperties": false,
  "not": [
    {
      "required": ["_og"]
    }
  ],
  "properties": {
    "_id": {"type": "string"},
    "Title": {"type": "string"}
  }
}
```

Le schéma de contrôle interdit que l’unité archivistique contienne des éléments de type géolocalisation (geo_point).

```
{
  "$schema": "http://vitam-json-schema.org/draft-04/schema#",
  "id": "http://example.com/root.json",
  "type": "object",
  "additionalProperties": false,
  "not": [
    {
      "type": "geo_point"
    }
  ],
  "properties": {
    "_id": {"type": "string"},
    "Title": {"type": "string"}
  }
}
```

```
}  
}
```

5.2.1.2. Propriétés d’un schéma de contrôle

Après l’en-tête, viennent les propriétés associées au schéma de contrôle. Ces propriétés correspondent à une liste d’éléments ou clés que l’on souhaite contrôler.

Si certains éléments du SEDA ne doivent pas faire l’objet d’un contrôle, il n’est pas nécessaire de les référencer dans le profil d’unité archivistique.

Généralités

Un élément est désigné par un **nom**, qui peut être explicité par un titre et/ou une description, tous deux facultatifs et devant obligatoirement contenir des éléments textuels.

Par exemple, l’élément suivant se nomme Description. Il dispose d’un titre et d’une description apportant des précisions sur son origine.

```
"Description": {  
  "title": "Description",  
  "description": "Element du SEDA, faisant partie du groupe DescriptionGroup."  
}
```

Mais chaque élément, propriété ou clé se définit surtout *a minima* par son nom et son **type**. On distingue plusieurs types possibles :

- string : texte ;
- number : nombre, entier ou décimal ;
- boolean : booléen dont la valeur est true ou false ;
- object : objet ;
- array : liste ou tableau de valeurs textuelles ;
- null : néant. L’élément peut être vide.

Un élément peut avoir plusieurs types. L’unité archivistique contenant un élément définissant plusieurs types sera valide si elle correspond à l’un des types déclarés.

Par exemple, l’élément intitulé City attend une valeur textuelle, tandis que l’élément Description soit est vide, soit requiert une valeur textuelle ou une liste de valeurs textuelles.

```
"City": {  
  "type": "string"  
}  
  
"Description": {  
  "description": "Element du SEDA, faisant partie du groupe DescriptionGroup. Peut être vide, répété (sous forme de tableau) et son contenu doit être textuel",  
  "type": [  
    "null",  
    "string",  
    "array"  
  ]  
}
```

```
}
```

La solution logicielle Vitam définit les types de ces propriétés dans l’ontologie.

Par analogie au SEDA et au langage XML, il convient de prêter attention aux éléments suivants :

- sera qualifié en objet un élément contenant des sous-éléments, par exemple : Management, Writer, Keyword ;
- sera qualifié en tableau (ou array) un élément répétable, tel que Tag ou OriginatingAgencyArchiveUnitIdentifiant ;
- certains éléments du SEDA (PreventInheritance, NeedAuthorization, NeedReassessingAuthorization) doivent contenir un booléen ;
- seront qualifiés de « null » des éléments pour lesquels le SEDA n’impose pas qu’ils soient obligatoirement renseignés, par exemple : BirthName ou Description. Néanmoins, dans ces cas-là, la solution logicielle Vitam proposera un autre type (texte par exemple).

Un élément peut également contenir les règles suivantes :

- **enum** : liste, permettant de signaler les valeurs autorisées. Au moins, l’une d’entre elle devra être présente pour se conformer au profil d’unité archivistique.

L’élément DescriptionLevel requiert l’utilisation d’une valeur égale à : RecordGrp, SubGrp ou File. Si l’unité archivistique contient une autre valeur, elle ne sera pas valide.

```
"DescriptionLevel": {  
  "type": "string",  
  "enum": [  
    "RecordGrp",  
    "SubGrp",  
    "File"  
  ]  
}
```

- **pattern** : texte exprimant une expression régulière. Le type date n’existait pas dans le format JSON, cette règle est notamment utilisée pour préciser le format de date attendu.

L’élément SigningTime attend une valeur correspondant à une expression régulière. Cette dernière désigne trois séries de chiffres compris entre 0 et 9, une série de 4 chiffres et deux de 2 chiffres, séparés par des tirets et se traduit par une date de type AAAA-MM-JJ.

```
"SigningTime": {  
  "description": "pattern: Date",  
  "anyOf": [  
    {  
      "type": "string",  
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"  
    }  
  ]  
}
```


- **anyOf** : liste imposant qu’au moins l’une des configurations déclarées soit vérifiée, quelle qu’elle soit.

Dans l’exemple suivant, pour se conformer au profil, l’unité archivistique devra contenir soit une valeur correspondant à l’expression régulière AAAA-MM-JJ suivie des secondes, soit le seul DataObjectReference, soit les trois éléments ensemble.

```
"BirthDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}[Z]?$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
}
```

- **allOf** : liste d’éléments devant être vérifiés. Tous les éléments doivent faire l’objet d’un contrôle.

L’unité archivistique devra contenir obligatoirement les éléments SystemId, Title et DataObjectReference. Si l’un d’eux manque, il ne sera pas conforme au profil d’unité archivistique.

```
{
  "$schema": "http://vitam-json-schema.org/draft-04/schema#",
  "id": "http://example.com/root.json",
  "type": "object",
  "additionalProperties": false,
  "allOf": [
    {
      "required": ["_og"],
      "required": ["_id", "Title"]
    }
  ],
  "properties": {
    "_id": {"type": "string"},
    "_og": {"type": "string"},
    "Title": {"type": "string"}
  }
}
```

- **oneOf** : liste imposant que les données soient exactement conformes à l’une des configurations proposées.

Le schéma de contrôle impose que l’unité archivistique contienne soit un SystemId et un Title, soit un DataObjectReference. En aucun cas, elle ne pourra contenir les trois éléments ensemble.

```
{
  "$schema": "http://vitam-json-schema.org/draft-04/schema#",
  "id": "http://example.com/root.json",
  "type": "object",
  "additionalProperties": false,
```

```
"oneOf": [
  {
    "required": ["_og"],
    "required": ["_id", "Title"]
  }
],
"properties": {
  "_id": {"type": "string"},
  "_og": {"type": "string"},
  "Title": {"type": "string"}
}
}
```

- **not** : liste déclarant des éléments ne devant pas être présent dans l’unité archivistique déclarant le profil d’unité archivistique ou excluant un paramétrage en particulier.

Le schéma de contrôle interdit que l’unité archivistique contienne un DataObjectReference.

```
{
  "$schema": "http://vitam-json-schema.org/draft-04/schema#",
  "id": "http://example.com/root.json",
  "type": "object",
  "additionalProperties": false,
  "not": [
    {
      "required": ["_og"]
    }
  ],
  "properties": {
    "_id": {"type": "string"},
    "Title": {"type": "string"}
  }
}
```

Le schéma de contrôle interdit que l’unité archivistique contienne des éléments de type géolocalisation (geo_point).

```
{
  "$schema": "http://vitam-json-schema.org/draft-04/schema#",
  "id": "http://example.com/root.json",
  "type": "object",
  "additionalProperties": false,
  "not": [
    {
      "type": "geo_point"
    }
  ],
  "properties": {
    "_id": {"type": "string"},
    "Title": {"type": "string"}
  }
}
```

5.2.1.3. Particularités de chaque type

Plusieurs types d’éléments présentent des spécificités permettant d’affiner le paramétrage de

celui-ci.

Type String

Un élément de type « string », correspondant à une chaîne de caractère, peut contenir les règles suivantes :

- **minLength, maxLength** : nombres entiers permettant de spécifier le minimum de caractères attendus comme valeur dans un champ et le maximum de caractères attendus dans un champ. Cette règle est notamment utile pour forcer à ce qu’un champ ait au moins une valeur renseignée (ex. champ Title du manifeste)

L’élément DescriptionLanguage requiert l’utilisation d’une valeur ayant un minimum de 1 caractère. Si, dans le manifeste, le champ est vide (aucune valeur n’est renseignée), l’unité archivistique ne sera pas validée.

```
"DescriptionLanguage": {
  "description": "End of LanguageGroup",
  "type": "string",
  "minLength": "1"
},
```

- **enum** : liste, permettant de signaler les valeurs autorisées. Au moins, l’une d’entre elle devra être présente pour se conformer au profil d’unité archivistique.

L’élément DescriptionLevel requiert l’utilisation d’une valeur égale à : RecordGrp, SubGrp ou File. Si l’unité archivistique contient une autre valeur, elle ne sera pas valide.

```
"DescriptionLevel": {
  "type": "string",
  "enum": [
    "RecordGrp",
    "SubGrp",
    "File"
  ]
}
```

- **pattern** : texte exprimant une expression régulière. Le type date n’existant pas dans le format JSON, cette règle est notamment utilisée pour préciser le format de date attendu.

L’élément SigningTime attend une valeur correspondant à une expression régulière. Cette dernière désigne trois séries de chiffres compris entre 0 et 9, une série de 4 chiffres et deux de 2 chiffres, séparés par des tirets et se traduit par une date de type AAAA-MM-JJ.

```
"SigningTime": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
    }
  ]
}
```

- **format** : liste permettant d’effectuer des contrôles prédéfinis pour un certain nombre de valeurs, notamment une date (vérification de la conformité de la valeur saisie par rapport à la RFC3339) ou une adresse électronique (vérification de la conformité de la valeur saisie par rapport à la RFC5322)

L’élément EventDateTime attend une valeur correspondant à une date conforme à la RFC3339, c’est-à-dire se présentant sous la forme AAAA-MM-JJThh:mm:ss. Tout unité archivistique déclarant dans l’élément EventDateTime une date non exprimée sous ce format ne sera pas validée.

```
"EventDateTime": {  
  "description": "pattern: Date",  
  "type": "string",  
  "format": "date-time"  
},
```

Type Number

Un élément de type « number », correspondant à une valeur numérique, peut contenir les règles suivantes :

- **minimum, maximum** : nombres entiers permettant de spécifier la valeur minimale et la valeur maximale attendues comme valeur dans un champ numérique.

L’élément ChildrenNumber (exemple fictif) attend une valeur numérique comprise entre 0 et 22. Toute unité archivistique déclarant une valeur supérieure à 22 ne sera pas validée. En revanche, une unité archivistique déclarant une valeur de 0 ou de 22 sera acceptée.

```
"ChildrenNumber": {  
  "type": "string",  
  "minimum": 0,  
  "maximum": 22  
}
```

- **exclusiveMinimum, exclusiveMaximum** : booléen dont la valeur est « true » (en minuscules) si on veut que la valeur attendue dans un champ numérique soit strictement supérieur à la valeur minimale définie ou strictement inférieure à la valeur supérieure définie.

L’élément ChildrenNumber (exemple fictif) attend une valeur numérique strictement comprise entre 0 et 22. Toute unité archivistique déclarant une valeur égale à 0, à 22 ou supérieure à 22 ne sera pas validée. En revanche, une unité archivistique déclarant une valeur de 1 ou de 21 sera acceptée.

```
"ChildrenNumber": {  
  "type": "string",  
  "minimum": 0,  
  "exclusiveMinimum": true,  
  "maximum": 22,  
  "exclusiveMaximum": true,  
}
```

Type Boolean

Un élément de type « boolean » doit nécessairement déclarer soit la valeur « true », soit la valeur « false », qui doit obligatoirement être en minuscule.

Type Array

Les éléments de type « array » sont les plus complexes à paramétrer dans un schéma de contrôle. Ils correspondent à une liste ou un tableau de valeurs textuelles (les **items**).

- **items** : entrée unitaire dans une liste ou un tableau de valeur textuelle.
 - cas où il n’y a qu’une seule entrée dans la liste ou le tableau :

L’élément OriginatingAgencyIdentifier est doté d’un type « array » dans le schéma de contrôle d’une unité archivistique, étant donné que, dans le schéma SEDA, sa cardinalité est de 0-n. Étant donné que l’élément ne correspond qu’à un seul champ dans le schéma SEDA, la liste de type « array » dans le schéma de contrôle ne déclare qu’un seul item, de type « string » ou « number »

```
"OriginatingAgencyArchiveUnitIdentifier": {
  "description": "Part of IdentifierGroup",
  "type": "array",
  "items": {
    "type":
      "string",
      "number"
  }
}
```

- cas où il y a plusieurs entrées dans la liste ou le tableau :

L’élément Event est doté d’un type « array » dans le schéma de contrôle d’une unité archivistique, étant donné que, dans le schéma SEDA, il s’agit d’un bloc de plusieurs métadonnées et dont la cardinalité est de 0-n. Étant donné que l’élément comprend plusieurs métadonnées à décrire, la liste de type « array » dans le schéma de contrôle doit déclarer plusieurs items. Le type sera donc « object », avec description ensuite, via les propriétés, des différents éléments composant l’élément Event.

```
"Event": {
  "type": [
    "array",
    "object"
  ],
  "items": {
    "type": "object",
    "properties": {
      "EventIdentifier": {
        "description": "NonEmptyTokenType",
        "type": "string"
      },
      "EventType": {
        "description": "NonEmptyTokenType",
        "type": "string"
      }
    }
  }
}
```

```

    },
    "EventDateTime": {
      "description": "pattern: Date",
      "anyOf": [
        {
          "type": "string",
          "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
        },
        {
          "type": "string",
          "format": "date-time"
        },
        {
          "type": "string",
          "format": "date-time-vitam"
        }
      ]
    },
    "EventDetail": {
      "description": "TextType",
      "type": "string"
    }
  }
}
}

```

- **minItems, maxItems** : nombres entiers permettant de spécifier le nombre minimal et le nombre maximal d’entrées attendues dans le tableau ou la liste de valeurs
 - cas où il n’y a qu’une seule entrée dans la liste ou le tableau :

La métadonnée OriginatingAgencyIdentifier pourra être absente ou présente une fois dans le bordereau. Une unité archivistique qui déclarera plus de 2 fois la métadonnée OriginatingAgencyIdentifier ne sera pas acceptée.

```

"OriginatingAgencyArchiveUnitIdentifier": {
  "description": "Part of IdentifierGroup",
  "type": "array",
  "minItems": "0",
  "maxItems": "1",
  "items": {
    "type":
      "string",
      "number"
  }
}
}

```

- cas où il y a plusieurs entrées dans la liste ou le tableau :

Le bloc Event pourra être absent ou présent entre 1 et 5 fois dans le bordereau. Une unité archivistique qui déclarera plus de 5 blocs Event ne sera pas acceptée.

```
"Event": {
  "type": [
    "array",
    "object"
  ],
  "minItems": "0",
  "maxItems": "5",
  "items": {
    "type": "object",
    "properties": {
      "EventIdentifier": {
        "description": "NonEmptyTokenType",
        "type": "string"
      },
      "EventType": {
        "description": "NonEmptyTokenType",
        "type": "string"
      },
      "EventDateTime": {
        "description": "pattern: Date",
        "anyOf": [
          {
            "type": "string",
            "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
          },
          {
            "type": "string",
            "format": "date-time"
          },
          {
            "type": "string",
            "format": "date-time-vitam"
          }
        ]
      },
      "EventDetail": {
        "description": "TextType",
        "type": "string"
      }
    }
  }
}
```

5.2.2. Comment exprimer les besoins métier dans un schéma de contrôle

5.2.2.1. Contexte d'utilisation d'une unité archivistique

A développer

5.2.2.2. Règles de gestion d’une unité archivistique

A développer

5.2.2.3. Métadonnées descriptives d’une unité archivistique

A développer

5.2.3. Validation et documentation d’un schéma de contrôle

5.2.4. Corrections d’un profil d’unité archivistique

5.2.5. Documentation d’un profil d’unité archivistique

Il est fortement recommandé de documenter le profil d’unité archivistique réalisé en réalisant sa propre modélisation, au moyen d’un schéma UML et d’un tableau récapitulatif :

Élément	Cardinalité	Type	Définitions et commentaires	Exemple	Obligatoire / Recommandé
DescriptionLevel	1.1	token	Pièce		
Title	1.1	string	Journal des transmissions		
...					

5.2.6. Réalisation d’un bordereau de transfert conforme au profil d’unité archivistique

Il est conseillé de réaliser un bordereau de transfert contenant une unité archivistique conforme au profil d’unité archivistique ou de demander au prestataire en charge de concevoir des exports de le faire.

Ce travail permet de :

- vérifier la structuration des données, et, le cas échéant, de corriger le profil d’unité archivistique ;
- vérifier la conformité au SEDA du bordereau de transfert conforme au profil d’archivage.

6. Conseils de mise en œuvre

À l’issue de cette première phase de réalisation de fonctionnalités concernant les profils d’unités archivistiques, l’équipe projet Vitam est en mesure de fournir quelques recommandations de mise en œuvre :

Intitulé	Description	Niveau de recommandation
Élaboration d’une notice descriptive de profil d’unité archivistique		
Import des notices descriptives	Afin d’optimiser la traçabilité de la création des différentes notices, il est recommandé de créer ces dernières une par une.	Recommandé
Modification des profils	La solution logicielle Vitam permet de modifier un profil d’unité archivistique et son schéma de contrôle . Cette action n’est pas recommandée quand le profil d’unité archivistique est interdit quand au moins une unité archivistique déclare ce profil dans le système	Non recommandé
Désactivation des profils	La solution logicielle Vitam permet de désactiver un profil d’unité archivistique. Cette action n’est pas recommandée quand le profil d’unité archivistique a déjà été utilisé en transfert comme en mise à jour Avant de désactiver un profil, il est recommandé de vérifier qu’aucun transfert nécessitant un contrôle par rapport à celui-ci n’est en cours ou programmé ou qu’aucune unité archivistique déjà présente dans le système ne déclare ce profil	Non recommandé
Besoin d’un profil d’unité archivistique		

Annexe 1 : Schéma de contrôle par défaut dans la solution logicielle Vitam (état au 16/05/2018)

```
{
  "$schema": "http://vitam-json-schema.org/draft-04/schema#",
  "id": "http://example.com/root.json",
  "type": "object",
  "additionalProperties": true,
  "anyOf": [
    {
      "required": [
        "_id",
        "Management",
        "DescriptionLevel",
        "Title"
      ]
    },
    {
      "required": [
        "_id",
        "Management",
        "DescriptionLevel",
        "Title_"
      ]
    }
  ],
  "properties": {
    "_id": {
      "type": "string"
    },
    "_og": {
      "type": "string"
    },
    "Management": {
      "id": "http://example.com/root.json/ArchiveUnit/Management",
      "additionalProperties": true,
      "properties": {
        "AccessRule": {
          "type": "object",
          "id": "http://example.com/root.json/ArchiveUnit/Management/AccessRule",
          "properties": {
            "Rules": {
              "type": "array",
              "id": "http://example.com/root.json/ArchiveUnit/Management/AccessRule/Rules",
              "description": "An explanation about the purpose of this instance.",
              "items": {
                "id": "http://example.com/root.json/ArchiveUnit/Management/AccessRule/Rules/Item/0",
                "type": "object",
                "properties": {
                  "Rule": {
                    "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
                    "title": "The Rule schema.",
                    "type": "string"
                  },
                  "StartDate": {
```

```

    "title": "The Startdate schema.",
    "description": "pattern: Date",
    "anyOf": [
      {
        "type": "string",
        "pattern": "^[0-8]{1}[0-9]{3}-[0-9]{2}-[0-9]{2}"
      }
    ]
  },
  "EndDate": {
    "title": "The Enddate schema.",
    "description": "pattern: Date",
    "anyOf": [
      {
        "type": "string",
        "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
      }
    ]
  }
}
},
"Inheritance": {
  "id": "http://example.com/root.json/ArchiveUnit/Management/AccessRule/Inheritance",
  "type": "object",
  "properties": {
    "PreventInheritance": {
      "id":
"http://example.com/root.json/ArchiveUnit/Management/AccessRule/Inheritance/PreventInheritance",
      "description": "An explanation about the purpose of this instance.",
      "title": "The PreventInheritance schema.",
      "type": [
        "string",
        "boolean"
      ]
    },
    "PreventRulesId": {
      "type": "array",
      "id":
"http://example.com/root.json/ArchiveUnit/Management/AccessRule/Inheritance/PreventRulesId",
      "items": {
        "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
        "title": "The PreventRulesId 0 schema.",
        "type": "string"
      }
    }
  }
},
"anyOf": [
  {
    "required": [
      "PreventInheritance"
    ]
  },
  {
    "required": [
      "PreventRulesId"
    ]
  }
]
}

```

```

    ]
  }
}
},
"AppraisalRule": {
  "type": "object",
  "id": "http://example.com/root.json/ArchiveUnit/Management/AppraisalRule",
  "properties": {
    "Rules": {
      "type": "array",
      "id": "http://example.com/root.json/ArchiveUnit/Management/AppraisalRule/Rules",
      "description": "An explanation about the purpose of this instance.",
      "items": {
        "id": "http://example.com/root.json/ArchiveUnit/Management/AppraisalRule/Rules/Item/0",
        "type": "object",
        "properties": {
          "Rule": {
            "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
            "title": "The Rule schema.",
            "type": "string"
          },
          "StartDate": {
            "title": "The Startdate schema.",
            "description": "pattern: Date",
            "anyOf": [
              {
                "type": "string",
                "pattern": "^[0-8]{1}[0-9]{3}-[0-9]{2}-[0-9]{2}"
              }
            ]
          },
          "EndDate": {
            "title": "The Enddate schema.",
            "description": "pattern: Date",
            "anyOf": [
              {
                "type": "string",
                "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
              }
            ]
          }
        }
      }
    },
    "FinalAction": {
      "enum": [
        "Keep",
        "Destroy"
      ]
    },
    "Inheritance": {
      "id": "http://example.com/root.json/ArchiveUnit/Management/AppraisalRule/Inheritance",
      "type": "object",
      "properties": {
        "PreventInheritance": {
          "id":
"http://example.com/root.json/ArchiveUnit/Management/AppraisalRule/Inheritance/PreventInheritance",
          "description": "An explanation about the purpose of this instance.",

```

```
"title": "The PreventInheritance schema.",
"type": [
  "string",
  "boolean"
]
},
"PreventRulesId": {
  "type": "array",
  "id":
"http://example.com/root.json/ArchiveUnit/Management/AppraisalRule/Inheritance/PreventRulesId",
  "items": {
    "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
    "title": "The PreventRulesId 0 schema.",
    "type": "string"
  }
},
"anyOf": [
  {
    "required": [
      "PreventInheritance"
    ]
  },
  {
    "required": [
      "PreventRulesId"
    ]
  }
]
},
"ArchiveUnit": {
  "id": "http://example.com/root.json/ArchiveUnit/Management/ArchiveUnit",
  "properties": {
    "ArchiveUnitRefId": {
      "description": "An explanation about the purpose of this instance.",
      "title": "The Archiveunitrefid schema.",
      "type": "string"
    }
  },
  "type": "object"
},
"ClassificationRule": {
  "type": "object",
  "id": "http://example.com/root.json/ArchiveUnit/Management/ClassificationRule",
  "properties": {
    "Rules": {
      "type": "array",
      "id": "http://example.com/root.json/ArchiveUnit/Management/ClassificationRule/Rules",
      "description": "An explanation about the purpose of this instance.",
      "items": {
        "id": "http://example.com/root.json/ArchiveUnit/Management/ClassificationRule/Rules/Item/0",
        "type": "object",
        "properties": {
          "Rule": {
            "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
            "title": "The Rule schema.",
```

```

    "type": "string"
  },
  "StartDate": {
    "title": "The Startdate schema.",
    "description": "pattern: Date",
    "anyOf": [
      {
        "type": "string",
        "pattern": "^([0-8]{1}[0-9]{3}-[0-9]{2}-[0-9]{2})"
      }
    ]
  },
  "EndDate": {
    "title": "The Enddate schema.",
    "description": "pattern: Date",
    "anyOf": [
      {
        "type": "string",
        "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
      }
    ]
  },
  "ClassificationReassessingDate": {
    "anyOf": [
      {
        "type": "string",
        "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
      },
      {
        "type": "string",
        "format": "date-time"
      }
    ]
  },
  "NeedReassessingAuthorization": {
    "type": "boolean"
  }
},
"ClassificationLevel": {
  "type": "string"
},
"ClassificationOwner": {
  "type": "string"
},
"Inheritance": {
  "id": "http://example.com/root.json/ArchiveUnit/Management/ClassificationRule/Inheritance",
  "type": "object",
  "properties": {
    "PreventInheritance": {
      "id":
"http://example.com/root.json/ArchiveUnit/Management/ClassificationRule/Inheritance/PreventInheritance",
      "description": "An explanation about the purpose of this instance.",
      "title": "The PreventInheritance schema.",
      "type": [
        "string",
        "boolean"
      ]
    }
  }
}

```

```

    ]
  },
  "PreventRulesId": {
    "type": "array",
    "id":
"http://example.com/root.json/ArchiveUnit/Management/ClassificationRule/Inheritance/PreventRulesId",
    "items": {
      "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
      "title": "The PreventRulesId 0 schema.",
      "type": "string"
    }
  }
},
"anyOf": [
  {
    "required": [
      "PreventInheritance"
    ]
  },
  {
    "required": [
      "PreventRulesId"
    ]
  }
]
},
      "anyOf": [ {"required":["ClassificationLevel", "ClassificationOwner"]}
],
"DisseminationRule": {
  "type": "object",
  "id": "http://example.com/root.json/ArchiveUnit/Management/DisseminationRule",
  "properties": {
    "Rules": {
      "type": "array",
      "id": "http://example.com/root.json/ArchiveUnit/Management/DisseminationRule/Rules",
      "description": "An explanation about the purpose of this instance.",
      "items": {
        "id": "http://example.com/root.json/ArchiveUnit/Management/DisseminationRule/Rules/Item/0",
        "type": "object",
        "properties": {
          "Rule": {
            "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
            "title": "The Rule schema.",
            "type": "string"
          },
          "StartDate": {
            "title": "The Startdate schema.",
            "description": "pattern: Date",
            "anyOf": [
              {
                "type": "string",
                "pattern": "^[0-8]{1}[0-9]{3}-[0-9]{2}-[0-9]{2}"
              }
            ]
          },
          "EndDate": {
            "title": "The Enddate schema.",

```

```

    "description": "pattern: Date",
    "anyOf": [
      {
        "type": "string",
        "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
      }
    ]
  }
}
},
"Inheritance": {
  "id": "http://example.com/root.json/ArchiveUnit/Management/DisseminationRule/Inheritance",
  "type": "object",
  "properties": {
    "PreventInheritance": {
      "id":
"http://example.com/root.json/ArchiveUnit/Management/DisseminationRule/Inheritance/PreventInheritance",
      "description": "An explanation about the purpose of this instance.",
      "title": "The PreventInheritance schema.",
      "type": [
        "string",
        "boolean"
      ]
    },
    "PreventRulesId": {
      "type": "array",
      "id":
"http://example.com/root.json/ArchiveUnit/Management/DisseminationRule/Inheritance/PreventRulesId",
      "items": {
        "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
        "title": "The PreventRulesId 0 schema.",
        "type": "string"
      }
    }
  },
  "anyOf": [
    {
      "required": [
        "PreventInheritance"
      ]
    },
    {
      "required": [
        "PreventRulesId"
      ]
    }
  ]
}
},
"NeedAuthorization": {
  "description": "An explanation about the purpose of this instance.",
  "title": "The Needauthorization schema.",
  "type": "boolean"
},
"ReuseRule": {
  "type": "object",

```



```
"id": "http://example.com/root.json/ArchiveUnit/Management/ReuseRule",
"properties": {
  "Rules": {
    "type": "array",
    "id": "http://example.com/root.json/ArchiveUnit/Management/ReuseRule/Rules",
    "description": "An explanation about the purpose of this instance.",
    "items": {
      "id": "http://example.com/root.json/ArchiveUnit/Management/ReuseRule/Rules/Item/0",
      "type": "object",
      "properties": {
        "Rule": {
          "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
          "title": "The Rule schema.",
          "type": "string"
        },
        "StartDate": {
          "title": "The Startdate schema.",
          "description": "pattern: Date",
          "anyOf": [
            {
              "type": "string",
              "pattern": "^[0-8]{1}[0-9]{3}-[0-9]{2}-[0-9]{2}"
            }
          ]
        },
        "EndDate": {
          "title": "The Enddate schema.",
          "description": "pattern: Date",
          "anyOf": [
            {
              "type": "string",
              "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
            }
          ]
        }
      }
    }
  },
  "Inheritance": {
    "id": "http://example.com/root.json/ArchiveUnit/Management/ReuseRule/Inheritance",
    "type": "object",
    "properties": {
      "PreventInheritance": {
        "id":
"http://example.com/root.json/ArchiveUnit/Management/ReuseRule/Inheritance/PreventInheritance",
        "description": "An explanation about the purpose of this instance.",
        "title": "The PreventInheritance schema.",
        "type": [
          "string",
          "boolean"
        ]
      },
      "PreventRulesId": {
        "type": "array",
        "id":
"http://example.com/root.json/ArchiveUnit/Management/ReuseRule/Inheritance/PreventRulesId",
        "items": {
          "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
```

```

    "title": "The PreventRulesId 0 schema.",
    "type": "string"
  }
}
},
"anyOf": [
  {
    "required": [
      "PreventInheritance"
    ]
  },
  {
    "required": [
      "PreventRulesId"
    ]
  }
]
}
},
"StorageRule": {
  "type": "object",
  "id": "http://example.com/root.json/ArchiveUnit/Management/StorageRule",
  "properties": {
    "Rules": {
      "type": "array",
      "id": "http://example.com/root.json/ArchiveUnit/Management/StorageRule/Rules",
      "description": "An explanation about the purpose of this instance.",
      "items": {
        "id": "http://example.com/root.json/ArchiveUnit/Management/StorageRule/Rules/Item/0",
        "type": "object",
        "properties": {
          "Rule": {
            "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
            "title": "The Rule schema.",
            "type": "string"
          },
          "StartDate": {
            "title": "The Startdate schema.",
            "description": "pattern: Date",
            "anyOf": [
              {
                "type": "string",
                "pattern": "[0-8]{1}[0-9]{3}-[0-9]{2}-[0-9]{2}"
              }
            ]
          },
          "EndDate": {
            "title": "The Enddate schema.",
            "description": "pattern: Date",
            "anyOf": [
              {
                "type": "string",
                "pattern": "[0-9]{4}-[0-9]{2}-[0-9]{2}"
              }
            ]
          }
        }
      }
    }
  }
}

```

```

    }
  },
  "FinalAction": {
    "enum": [
      "RestrictAccess",
      "Transfer",
      "Copy"
    ]
  },
  "Inheritance": {
    "id": "http://example.com/root.json/ArchiveUnit/Management/StorageRule/Inheritance",
    "type": "object",
    "properties": {
      "PreventInheritance": {
        "id":
"http://example.com/root.json/ArchiveUnit/Management/StorageRule/Inheritance/PreventInheritance",
        "description": "An explanation about the purpose of this instance.",
        "title": "The PreventInheritance schema.",
        "type": [
          "string",
          "boolean"
        ]
      },
      "PreventRulesId": {
        "type": "array",
        "id":
"http://example.com/root.json/ArchiveUnit/Management/StorageRule/Inheritance/PreventRulesId",
        "items": {
          "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
          "title": "The PreventRulesId 0 schema.",
          "type": "string"
        }
      }
    }
  },
  "anyOf": [
    {
      "required": [
        "PreventInheritance"
      ]
    },
    {
      "required": [
        "PreventRulesId"
      ]
    }
  ]
}
},
"UpdateOperation": {
  "id": "http://example.com/root.json/ArchiveUnit/Management/UpdateOperation",
  "properties": {
    "SystemId": {
      "description": "An explanation about the purpose of this instance.",
      "title": "The Systemid schema.",
      "type": "string"
    },
    "ToDelete": {

```

```
"id": "http://example.com/root.json/ArchiveUnit/Management/UpdateOperation/ToDelete",
"properties": {
  "ArchiveUnitRefId": {
    "id":
"http://example.com/root.json/ArchiveUnit/Management/UpdateOperation/ToDelete/ArchiveUnitRefId",
    "items": {
      "description": "An explanation about the purpose of this instance.",
      "title": "The 0 schema.",
      "type": "string"
    },
    "type": "array"
  },
  "FullUpdate": {
    "description": "An explanation about the purpose of this instance.",
    "title": "The Fullupdate schema.",
    "type": "boolean"
  }
},
"type": "object"
}
},
"type": "object"
},
},
"DescriptionLevel": {
  "type": "string",
  "enum": [
    "Fonds",
    "Subfonds",
    "Class",
    "Collection",
    "Series",
    "Subseries",
    "RecordGrp",
    "SubGrp",
    "File",
    "Item"
  ]
},
"Title": {
  "description": "All TitleGroup",
  "type": [
    "string",
    "array",
    "number"
  ],
  "minLength": 1,
  "minItems": 1
},
"FilePlanPosition": {
  "description": "Start of IdentifierGroup",
  "type": "array",
  "items": {
    "type": "string"
  }
},
}
```

```
"SystemId": {
  "description": "Part of IdentifierGroup",
  "type": "array",
  "items": {
    "type": "string"
  }
},
"OriginatingSystemId": {
  "description": "Part of IdentifierGroup",
  "type": "array",
  "items": {
    "type": [
      "string",
      "number"
    ]
  }
},
"ArchivalAgencyArchiveUnitIdentifier": {
  "description": "Part of IdentifierGroup",
  "type": "array",
  "items": {
    "type": [
      "string",
      "number"
    ]
  }
},
"OriginatingAgencyArchiveUnitIdentifier": {
  "description": "Part of IdentifierGroup",
  "type": "array",
  "items": {
    "type": [
      "string",
      "number"
    ]
  }
},
"TransferringAgencyArchiveUnitIdentifier": {
  "description": "End of IdentifierGroup",
  "type": "array",
  "items": {
    "type": [
      "string",
      "number"
    ]
  }
},
>Description": {
  "description": "All of DescriptionGroup",
  "type": [
    "null",
    "string",
    "array"
  ]
},
"CustodialHistory": {
  "description": "All of CustodialHistoryGroup",
  "type": "object",
```

```
"properties": {
  "CustodialHistoryItem": {
    "type": "array",
    "items": {
      "type": "string"
    }
  },
  "CustodialHistoryFile": {
    "type": "object",
    "properties": {
      "DataObjectReferenceId": {
        "description": "xsd:IDREF",
        "type": "string"
      },
      "DataObjectGroupReferenceId": {
        "description": "xsd:IDREF",
        "type": "string"
      }
    }
  }
},
"required": [
  "CustodialHistoryItem"
],
"Type": {
  "description": "All of TypeGroup",
  "type": [
    "string",
    "object"
  ]
},
"DocumentType": {
  "description": "All of DocumentTypeGroup",
  "type": "string"
},
"Language": {
  "description": "Start of LanguageGroup",
  "type": "array",
  "items": {
    "type": "string"
  }
},
>DescriptionLanguage": {
  "description": "End of LanguageGroup",
  "type": "string"
},
>Status": {
  "description": "All of StatusGroup",
  "type": "string"
},
Version": {
  "description": "All of VersionGroup",
  "type": "string"
},
Tag": {
  "description": "Start of KeywordsGroup",
  "type": "array",
```

```
"items": {
  "type": "string"
}
},
"Keyword": {
  "description": "End of KeywordsGroup",
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "KeywordContent": {
        "type": "string"
      },
      "KeywordReference": {
        "type": "string"
      },
      "KeywordType": {
        "description": "How TODO ? listVersionID (attribute) + String: [@listVersionID + $, Only string]?",
        "enum": [
          "corpname",
          "famname",
          "geogname",
          "name",
          "occupation",
          "persname",
          "subject",
          "genreform",
          "function"
        ]
      }
    }
  }
},
"Coverage": {
  "type": "object",
  "properties": {
    "Spatial": {
      "type": [
        "string",
        "array",
        "number"
      ]
    }
  },
  "Temporal": {
    "type": [
      "string",
      "array",
      "number"
    ]
  },
  "Jurisdictional": {
    "type": [
      "string",
      "array",
      "number"
    ]
  }
}
```

```
},
"OriginatingAgency": {
  "description": "All of OriginatingAgencyGroup",
  "properties": {
    "Identifier": {
      "type": "string"
    },
    "OrganizationDescriptiveMetadata": {
      "additionalProperties": true,
      "type": "object"
    }
  },
  "type": "object"
},
"SubmissionAgency": {
  "description": "All of SubmissionAgencyGroup",
  "properties": {
    "Identifier": {
      "type": "string"
    },
    "OrganizationDescriptiveMetadata": {
      "additionalProperties": true,
      "type": "object"
    }
  },
  "type": "object"
},
"AuthorizedAgent": {
  "description": "All of AuthorizedAgentGroup",
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "FirstName": {
        "type": "string"
      },
      "BirthName": {
        "type": "string"
      },
      "GivenName": {
        "type": "string"
      },
      "Gender": {
        "type": "string"
      },
      "BirthDate": {
        "description": "pattern: Date",
        "anyOf": [
          {
            "type": "string",
            "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
          },
          {
            "type": "string",
            "format": "date-time"
          }
        ]
      }
    }
  }
},
```



```
"BirthPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"DeathDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"DeathPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
}
```

```
    }
  }
},
"Nationality": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Identifier": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Corpname": {
  "type": "string"
}
},
"anyOf": [
  {
    "anyOf": [
      {
        "required": [
          "FirstName"
        ]
      },
      {
        "required": [
          "BirthName"
        ]
      },
      {
        "required": [
          "GivenName"
        ]
      },
      {
        "required": [
          "Gender"
        ]
      },
      {
        "required": [
          "BirthDate"
        ]
      },
      {
        "required": [
          "DeathDate"
        ]
      },
      {
        "required": [
          "DeathPlace"
        ]
      },
      {

```

```
    "required": [
      "Nationality"
    ]
  }
]
},
{
  "anyOf": [
    {
      "required": [
        "Corpname"
      ]
    }
  ]
}
]
}
},
"Writer": {
  "type": [
    "array",
    "object"
  ],
  "items": {
    "type": "object",
    "properties": {
      "FirstName": {
        "type": [
          "string",
          "null"
        ]
      },
      "BirthName": {
        "type": [
          "string",
          "null"
        ]
      },
      "GivenName": {
        "type": [
          "string",
          "null"
        ]
      },
      "Gender": {
        "type": [
          "string",
          "null"
        ]
      },
      "BirthDate": {
        "description": "pattern: Date",
        "anyOf": [
          {
            "type": "string",
            "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}[Z]?$"
          },
          {

```

```
    "type": "string",
    "format": "date-time"
  }
]
},
"BirthPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"DeathDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2}[Z]?)$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"DeathPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    }
  },
}
```

```
"Region": {
  "type": "string"
},
"Country": {
  "type": "string"
}
},
"Nationality": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Identifier": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Function": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Activity": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Position": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Role": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Mandate": {
  "type": "array",
  "items": {
    "type": "string"
  }
}
},
"Addressee": {
  "description": "All of AddresseeGroup (AudienceGroup)",
  "items": {
    "type": "object",
    "properties": {
```

```
"FirstName": {
  "type": [
    "string",
    "null"
  ]
},
"BirthName": {
  "type": [
    "string",
    "null"
  ]
},
"GivenName": {
  "type": [
    "string",
    "null"
  ]
},
"Gender": {
  "type": [
    "string",
    "null"
  ]
},
"BirthDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"BirthPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
}
```

```
    }
  }
},
"DeathDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"DeathPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"Nationality": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Identifier": {
  "items": {
    "type": "string"
  },
  "type": [
    "array",
    "string"
  ]
},
"Corpname": {
  "type": "string"
}
```

```
},
"anyOf": [
  {
    "anyOf": [
      {
        "required": [
          "FirstName"
        ]
      },
      {
        "required": [
          "BirthName"
        ]
      },
      {
        "required": [
          "GivenName"
        ]
      },
      {
        "required": [
          "Gender"
        ]
      },
      {
        "required": [
          "BirthDate"
        ]
      },
      {
        "required": [
          "DeathDate"
        ]
      },
      {
        "required": [
          "DeathPlace"
        ]
      },
      {
        "required": [
          "Nationality"
        ]
      }
    ]
  },
  {
    "anyOf": [
      {
        "required": [
          "Corpname"
        ]
      }
    ]
  }
]
},
},
```



```
"Recipient": {
  "description": "All of RecipientGroup (AudienceGroup)",
  "items": {
    "type": "object",
    "properties": {
      "FirstName": {
        "type": [
          "string",
          "null"
        ]
      },
      "BirthName": {
        "type": [
          "string",
          "null"
        ]
      },
      "GivenName": {
        "type": [
          "string",
          "null"
        ]
      },
      "Gender": {
        "type": [
          "string",
          "null"
        ]
      },
      "BirthDate": {
        "description": "pattern: Date",
        "anyOf": [
          {
            "type": "string",
            "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
          },
          {
            "type": "string",
            "format": "date-time"
          }
        ]
      },
      "BirthPlace": {
        "description": "LocationGroup",
        "type": "object",
        "properties": {
          "Geogname": {
            "type": "string"
          },
          "Address": {
            "type": "string"
          },
          "PostalCode": {
            "type": "string"
          },
          "City": {
            "type": "string"
          }
        }
      }
    }
  }
}
```

```
"Region": {
  "type": "string"
},
"Country": {
  "type": "string"
}
},
"DeathDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"DeathPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"Nationality": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Identifier": {
  "items": {
    "type": "string"
  },
  "type": [
    "array",
    "string"
  ]
}
```

```
]
},
"Corpname": {
  "type": "string"
}
},
"anyOf": [
  {
    "anyOf": [
      {
        "required": [
          "FirstName"
        ]
      },
      {
        "required": [
          "BirthName"
        ]
      },
      {
        "required": [
          "GivenName"
        ]
      },
      {
        "required": [
          "Gender"
        ]
      },
      {
        "required": [
          "BirthDate"
        ]
      },
      {
        "required": [
          "DeathDate"
        ]
      },
      {
        "required": [
          "DeathPlace"
        ]
      },
      {
        "required": [
          "Nationality"
        ]
      }
    ]
  }
],
{
  "anyOf": [
    {
      "required": [
        "Corpname"
      ]
    }
  ]
}
```

```
    ]
  }
]
},
"Source": {
  "description": "All of SourceGroup",
  "type": "string"
},
"RelatedObjectReference": {
  "description": "All of RelationGroup",
  "type": "object",
  "properties": {
    "IsVersionOf": {
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "ArchiveUnitRefId": {
            "description": "TYPE: xsd:IDREF",
            "type": "string"
          },
          "DataObjectReference": {
            "type": "object",
            "properties": {
              "DataObjectReferenceId": {
                "description": "TYPE: xsd:IDREF",
                "type": "string"
              },
              "DataObjectGroupReferenceId": {
                "description": "TYPE: xsd:IDREF",
                "type": "string"
              }
            }
          }
        }
      },
      "RepositoryArchiveUnitPID": {
        "description": "NonEmptyTokenType",
        "type": "string"
      },
      "RepositoryObjectPID": {
        "description": "NonEmptyTokenType",
        "type": "string"
      }
    }
  }
},
"Replaces": {
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "ArchiveUnitRefId": {
        "description": "TYPE: xsd:IDREF",
        "type": "string"
      },
      "DataObjectReference": {
        "type": "object",
        "properties": {
```

```
"DataObjectReferenceId": {
  "description": "TYPE: xsd:IDREF",
  "type": "string"
},
"DataObjectGroupReferenceId": {
  "description": "TYPE: xsd:IDREF",
  "type": "string"
}
},
"RepositoryArchiveUnitPID": {
  "description": "NonEmptyTokenType",
  "type": "string"
},
"RepositoryObjectPID": {
  "description": "NonEmptyTokenType",
  "type": "string"
}
}
},
"Requires": {
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "ArchiveUnitRefId": {
        "description": "TYPE: xsd:IDREF",
        "type": "string"
      },
      "DataObjectReference": {
        "type": "object",
        "properties": {
          "DataObjectReferenceId": {
            "description": "TYPE: xsd:IDREF",
            "type": "string"
          },
          "DataObjectGroupReferenceId": {
            "description": "TYPE: xsd:IDREF",
            "type": "string"
          }
        }
      }
    }
  },
  "RepositoryArchiveUnitPID": {
    "description": "NonEmptyTokenType",
    "type": "string"
  },
  "RepositoryObjectPID": {
    "description": "NonEmptyTokenType",
    "type": "string"
  }
}
},
"IsPartOf": {
  "type": "array",
  "items": {
    "type": "object",
```

```
"properties": {
  "ArchiveUnitRefId": {
    "description": "TYPE: xsd:IDREF",
    "type": "string"
  },
  "DataObjectReference": {
    "type": "object",
    "properties": {
      "DataObjectReferenceId": {
        "description": "TYPE: xsd:IDREF",
        "type": "string"
      },
      "DataObjectGroupReferenceId": {
        "description": "TYPE: xsd:IDREF",
        "type": "string"
      }
    }
  },
  "RepositoryArchiveUnitPID": {
    "description": "NonEmptyTokenType",
    "type": "string"
  },
  "RepositoryObjectPID": {
    "description": "NonEmptyTokenType",
    "type": "string"
  }
}
},
"References": {
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "ArchiveUnitRefId": {
        "description": "TYPE: xsd:IDREF",
        "type": "string"
      },
      "DataObjectReference": {
        "type": "object",
        "properties": {
          "DataObjectReferenceId": {
            "description": "TYPE: xsd:IDREF",
            "type": "string"
          },
          "DataObjectGroupReferenceId": {
            "description": "TYPE: xsd:IDREF",
            "type": "string"
          }
        }
      },
      "RepositoryArchiveUnitPID": {
        "description": "NonEmptyTokenType",
        "type": "string"
      },
      "RepositoryObjectPID": {
        "description": "NonEmptyTokenType",
        "type": "string"
      }
    }
  }
}
```

```
    }
  }
}
},
"CreatedDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    },
    {
      "type": "string",
      "format": "date-time-vitam"
    }
  ]
},
"TransactedDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    },
    {
      "type": "string",
      "format": "date-time-vitam"
    }
  ]
},
"AcquiredDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    },
    {
      "type": "string",
      "format": "date-time-vitam"
    }
  ]
},
"SentDate": {
```

```
"description": "pattern: Date",
"anyOf": [
  {
    "type": "string",
    "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
  },
  {
    "type": "string",
    "format": "date-time"
  },
  {
    "type": "string",
    "format": "date-time-vitam"
  }
]
},
"ReceivedDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    },
    {
      "type": "string",
      "format": "date-time-vitam"
    }
  ]
},
"RegisteredDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    },
    {
      "type": "string",
      "format": "date-time-vitam"
    }
  ]
},
"StartDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {

```



```
"type": "string",
"format": "date-time"
},
{
  "type": "string",
  "format": "date-time-vitam"
}
]
},
"EndDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    },
    {
      "type": "string",
      "format": "date-time-vitam"
    }
  ]
},
"Event": {
  "type": [
    "array",
    "object"
  ],
  "items": {
    "type": "object",
    "properties": {
      "EventIdentifier": {
        "description": "NonEmptyTokenType",
        "type": "string"
      },
      "EventType": {
        "description": "NonEmptyTokenType",
        "type": "string"
      },
      "EventDateTime": {
        "description": "pattern: Date",
        "anyOf": [
          {
            "type": "string",
            "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
          },
          {
            "type": "string",
            "format": "date-time"
          },
          {
            "type": "string",
            "format": "date-time-vitam"
          }
        ]
      }
    }
  ]
}
```

```
    },
    "EventDetail": {
      "description": "TextType",
      "type": "string"
    }
  }
},
"Signature": {
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "Signer": {
        "type": "array",
        "items": {
          "type": "object",
          "properties": {
            "FirstName": {
              "type": "string"
            },
            "BirthName": {
              "type": "string"
            },
            "FullName": {
              "type": "string"
            },
            "GivenName": {
              "type": "string"
            },
            "Gender": {
              "type": "string"
            },
            "BirthDate": {
              "description": "pattern: Date",
              "anyOf": [
                {
                  "type": "string",
                  "pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2})$"
                },
                {
                  "type": "string",
                  "format": "date-time"
                }
              ]
            },
            "BirthPlace": {
              "description": "LocationGroup",
              "type": "object",
              "properties": {
                "Geogname": {
                  "type": "string"
                },
                "Address": {
                  "type": "string"
                },
                "PostalCode": {
                  "type": "string"
                }
              }
            }
          }
        }
      }
    }
  }
}
```

```
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"DeathDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"DeathPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"Nationality": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Identifier": {
  "items": {
    "type": "string"
  }
}
```

```
  },
  "type": "array"
},
"Corpname": {
  "type": "string"
},
"SigningTime": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
    }
  ]
},
"Function": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Activity": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Position": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Role": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Mandate": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"anyOf": [
  {
    "anyOf": [
      {
        "required": [
          "FirstName"
        ]
      },
      {
        "required": [
          "BirthName"
        ]
      }
    ]
  }
]
```

```
},
{
  "required": [
    "FullName"
  ]
},
{
  "required": [
    "GivenName"
  ]
},
{
  "required": [
    "Gender"
  ]
},
{
  "required": [
    "BirthDate"
  ]
},
{
  "required": [
    "DeathDate"
  ]
},
{
  "required": [
    "DeathPlace"
  ]
},
{
  "required": [
    "Nationality"
  ]
}
]
},
{
  "anyOf": [
    {
      "required": [
        "Corpname"
      ]
    }
  ]
}
]
}
},
"Validator": {
  "type": "object",
  "properties": {
    "FirstName": {
      "type": "string"
    },
    "BirthName": {
      "type": "string"
    }
  }
}
```

```
},
"FullName": {
  "type": "string"
},
"GivenName": {
  "type": "string"
},
"Gender": {
  "type": "string"
},
"BirthDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"BirthPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"DeathDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
}
```

```
]
},
"DeathPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"Nationality": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Identifier": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Corpname": {
  "type": "string"
},
"ValidationTime": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
    }
  ]
},
"Function": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Activity": {
  "type": "array",
```

```
"items": {
  "type": "string"
},
"Position": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Role": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Mandate": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"anyOf": [
  {
    "anyOf": [
      {
        "required": [
          "FirstName"
        ]
      },
      {
        "required": [
          "BirthName"
        ]
      },
      {
        "required": [
          "FullName"
        ]
      },
      {
        "required": [
          "GivenName"
        ]
      },
      {
        "required": [
          "Gender"
        ]
      },
      {
        "required": [
          "BirthDate"
        ]
      },
      {
        "required": [
```



```

        "DeathDate"
      ]
    },
    {
      "required": [
        "DeathPlace"
      ]
    },
    {
      "required": [
        "Nationality"
      ]
    }
  ]
},
{
  "anyOf": [
    {
      "required": [
        "Corpname"
      ]
    }
  ]
}
]
},
"Masterdata": {
  "description": "Type: CodeType (NonEmptyTokenType + FIXME: some attributes, How to do)",
  "type": "string"
},
"ReferencedObject": {
  "type": "object",
  "properties": {
    "SignedObjectId": {
      "description": "xsd:IDREF",
      "type": "string"
    },
    "SignedObjectDigest": {
      "description": "String type: (FIXME: Check xsd:base64Binary OR xsd:hexBinary, can specify Format
for those ?) ∨ + FIXME: Attributes (How to do)",
      "type": "object"
    }
  }
}
}
}
},
"Gps": {
  "type": "object",
  "properties": {
    "GpsVersionID": {
      "type": "string"
    },
    "GpsAltitude": {
      "type": "integer"
    },
    "GpsAltitudeRef": {

```

```
"type": "string"  
},  
"GpsLatitude": {  
  "type": "string"  
},  
"GpsLatitudeRef": {  
  "type": "string"  
},  
"GpsLongitude": {  
  "type": "string"  
},  
"GpsLongitudeRef": {  
  "type": "string"  
},  
"GpsDateStamp": {  
  "type": "string"  
}  
}  
}
```

Annexe 2 : Exemple de notice détaillée d’un profil d’unité archivistique

Nota bene : les cas présentés ci-dessous n’ont pas vocation à être vraisemblables d’un point de vue métier. Ils visent simplement à vérifier la bonne mise en œuvre des mécanismes relatifs aux habilitations dans la solution logicielle Vitam.

```
{  
  "Name": "Facture",  
  "Description": "Document type d’une facture associée à un dossier de marché",  
  "Status": "ACTIVE",  
  "ControlSchema": "{}"  
}
```

Annexe 3 : Messages d’erreur

Nota bene : les cas présentés ci-dessous n’ont pas vocation à être exhaustif.

	Message retourné par la solution logicielle Vitam	Explication	Résolution
Erreur 1	<pre>{ "evDetTechData": { "validateJson": { "level": "error", "schema": { "loadingURI": "#", "pointer": "/properties/OriginatingSystemId", "instance": { "pointer": "/OriginatingSystemId", "domain": "validation", "keyword": "type", "message": "instance type (array) does not match any allowed primitive type (allowed: [integer, number, string])", "found": "array", "expected": ["integer", "number", "string"]} } } } } }</pre>		
Erreur 2	<pre>{ "ArchiveUnit": "ID26", "ArchiveUnitProfile": "AUP-000002", "evDetTechData": { "validateUnitReport": { "level": "error", "schema": { "loadingURI": "#", "pointer": "", "instance": { "pointer": "", "domain": "validation", "keyword": "additionalProperties", "message": "object instance has properties which are not allowed by the schema: [DataObjectReference, Management]", "unwanted": ["DataObjectReference", "Management"]} } } } } }</pre>	<p>L’en-tête du profil déclare la valeur « false » dans le champ <code>additionalProperties</code>. Or l’objet json correspondant au bloc <code>DataObjectReference</code> du schéma SEDA n’est pas déclaré dans le schéma de contrôle json et l’unité archivistique déclare un groupe d’objet dans le manifeste. Donc le schéma de l’unité archivistique n’est pas conforme à son schéma de contrôle</p>	<p>2 solutions :</p> <ul style="list-style-type: none"> • modifier la valeur du champ <code>additionalProperties</code> dans l’en-tête du schéma de contrôle • déclarer dans le schéma de contrôle l’objet correspondant au bloc <code>DataObjectReference</code> du schéma SEDA
Erreur 3	<pre>{ "ArchiveUnit": "ID3", "ArchiveUnitProfile": "AUP-000005", "evDetTechData": { "validateUnitReport": { "level": "warning", "schema": { "loadingURI": "#", "pointer": "/properties/Management/properties/AppraisalRule", "domain": "syntax", "message": "the following keywords are unknown and will be ignored:" } } } }</pre>	<p>Présence dans le manifeste d’une métadonnée non autorisée dans le schéma</p>	<p>3 solutions :</p> <ul style="list-style-type: none"> • modifier la valeur du champ <code>additionalProperties</code> dans l’en-tête du schéma de contrôle pour autoriser les métadonnées supplémentaires • déclarer dans le schéma de contrôle l’objet correspondant à

	<pre>[FinalAction]"",\ignored": ["FinalAction"], {"level":"error",\schema": {"loadingURI":"#\","pointer":\ "},\instance": {"pointer":"","\domain":"val idation",\keyword":"additional Properties",\message":"object instance has properties which are not allowed by the schema: ["ChildrenNumber"]",\unwa nted":["ChildrenNumber"]}]}</pre>		<p>la métadonnée ChildrenNumber</p> <ul style="list-style-type: none"> • modifier le manifeste du SIP pour retirer la métadonnée correspondante
Erreur 4	<pre>{"ArchiveUnit":"ID3",\ArchiveU nitProfile":"AUP- 000005",\evDetTechData":{"val idateUnitReport": [{"level":"warning",\schema" : {"loadingURI":"#\","pointer":\ "/properties/Management/properti es/AppraisalRule"},\domain":" syntax",\message":"the following keywords are unknown and will be ignored: [FinalAction]"",\ignored": ["FinalAction"], {"level":"error",\schema": {"loadingURI":"#\","pointer":\ "},\instance": {"pointer":"","\domain":"val idation",\keyword":"anyOf",\ message":"instance failed to match at least one required schema among 1",\nrSchemas":1,\reports": {"anyOf/0": [{"level":"error",\schema": {"loadingURI":"#\","pointer":\ "/anyOf/0"},\instance": {"pointer":"","\domain":"val idation",\keyword":"required", \message":"object has missing required properties (["ChildrenNumber"])",\req uired": ["ChildrenNumber",\Descriptio nLevel",\Management",\Recru itmentDate",\StartDate",\Title\ "],\missing": ["ChildrenNumber"]}]}}}]}</pre>	Absence d’une métadonnée attendue (ChildrenNumber)	Corriger le manifeste du SIP pour rajouter la métadonnée attendue
Erreur 5	<pre>{"ArchiveUnit":"ID3",\ArchiveU nitProfile":"AUP- 000005",\evDetTechData":{"val idateUnitReport": [{"level":"warning",\schema" :</pre>	La valeur du champ ChildrenNumber est incompatible avec le type déclarée dans le schéma	Corriger le manifeste du SIP pour rendre la valeur du champ ChildrenNumber compatible avec le type déclaré dans le schéma de contrôle

	<pre>{ "loadingURI": "#", "pointer": "/properties/Management/properties/AppraisalRule", "domain": "syntax", "message": "the following keywords are unknown and will be ignored: [FinalAction], ignored: [FinalAction]", "level": "error", "schema": { "loadingURI": "#", "pointer": "/properties/ChildrenNumber", "instance": { "pointer": "/ChildrenNumber", "domain": "validation", "keyword": "type", "message": "instance type (array) does not match any allowed primitive type (allowed: [integer, number])", "found": "array", "expected": [integer, number] }, "level": "warning", "schema": { "loadingURI": "#", "pointer": "/properties/Management/properties/AppraisalRule", "domain": "syntax", "message": "the following keywords are unknown and will be ignored: [FinalAction], ignored: [FinalAction]" } } }</pre>		

Pour les éléments propres au SEDA, le tableau suivant précise les types de certains d’entre eux :

	string	number	boolean	object	array	null
_id						
_og						
ArchiveProfile						
Management				x		
AccessRule				x		
AppraisalRule				x		
StorageRule				x		
ReuseRule				x		
ClassificationRule				x		
Rule	x					
StartDate	x ³					

Programme Vitam – Profils d'unité archivistique – v 2.0

EndDate	x ⁴					
FinalAction ⁵						
Inheritance				x		
PreventInheritance	x		x			
PreventRulesId					x ⁶	
Title	x				x	

3 Avec un pattern date

4 Idem.

5 Simple énumération

6 Tableau contenant des éléments de type string